
Modelling Fed-Batch Fermentation Processes: An Approach Based on Artificial Neural Networks

Eduardo Valente^{1,2}, Isabel Rocha³, and Miguel Rocha¹

¹ Dep. Informatics / CCTC - University of Minho
Campus de Gualtar, 4710-057 Braga - Portugal
mrocha@di.uminho.pt

² Dep. Engenharia Informática, ESTCB
Castelo Branco - Portugal
eduardo@est.ipcb.pt

³ IBB - Institute for Biotechnology and Bioengineering
Center of Biological Engineering - University of Minho
Campus de Gualtar, 4710-057 Braga - Portugal
irocha@deb.uminho.pt

Summary. Artificial Neural Networks (ANNs) have shown to be powerful tools for solving several problems which, due to their complexity, are extremely difficult to unravel with other methods. Their capabilities of massive parallel processing and learning from the environment make these structures ideal for prediction of nonlinear events. In this work, a set of computational tools are proposed, allowing researchers in Biotechnology to use ANNs for the modelling of fed-batch fermentation processes. The main task is to predict the values of kinetics parameters from the values of a set of state variables. The tools were validated with two case studies, showing the main functionalities of the application.

Keywords: Multilayer Perceptrons, Bioprocess Modelling, Biotechnology, Fed-batch fermentation processes, Bioinformatics.

1 Introduction

The productivity of the manufacturing processes is of paramount concern of any company that wishes to survive in the market. The strong competition leads to the search for new strategies and to the investment in alternatives to traditional methods. Under this scenario, the use of fermentation techniques is steadily growing as a solution to achieve the production of targets with high economic value such as recombinant proteins, antibiotics and amino-acids. Also, they have been replacing traditional processes in areas such as the production of bulk chemicals, given their low energy requirements and environmental costs. However, these processes are typically very complex, involving different transport phenomena, microbial components and biochemical reactions. The nonlinear behavior and time-varying properties limits the application of traditional control and optimization techniques to bioreactors.

A fermentation process can be described as any process that produces a specific product through the mass culture of a microorganism [13]. The yield, in this case, means increasing the cell reproduction rates, or the production of compounds from these cells, per unit of time. The factors influencing a fermentation process range from the physical and chemical conditions of the medium (temperature, pH, etc.), sources of energy (glucose, oxygen, light) and nutrients (nitrogen) supplied to the culture. Traditionally, empirical methods based on Monod equations have been adopted for the optimisation of these processes. Each strain of a microorganism has its own specificities and the way it is grown also gives rise to distinct culture behaviour.

The modelling task is essential to achieve the optimization of these processes, since it provides simulated data that are often difficult to measure directly from the culture. One important example is the kinetic behaviour, which depends on factors involving cellular catalysts, intracellular phenomena and characteristics of the population [9], among other complex aspects which makes its collection impossible to be carried out in real-time. Usually, the kinetics is estimated by the measurement of other variables of the process, which in most cases can only be obtained at the end of the experiments. Work in this field has led to the development of tools that attempt to address problems specific to a particular culture, with limited generalization capabilities. Another major problem is the complexity of the models, which use mathematical computations that require lots of processing resources [15, 16].

This work aims to demonstrate that Artificial Neural Networks (ANNs) are adequate to this task, due to their features of massive parallel computing and representation, ability to learn, adaptability and generalization [4]. ANNs, and in particular Multilayer Perceptrons (MLPs) used in this case, are mainly characterized by the topology of connections between neurons, the training algorithm and the activation function [3, 2]. The choice of which ANN will be used to represent a process depends on the characteristics of the problem in question [7]. There are several methods to reach the desired solution, that range from exhaustive search, to local optimization techniques and also the use of Evolutionary Algorithms [5, 14].

The main contribution of this work is the development of computational tools that allow a Biotechnology researcher, with very limited programming skills, to be able to model fermentation processes using ANNs. These will be used to predict the values of kinetic parameters from state variables. A number of tools will be provided to create ANN models from data, to evaluate a set of distinct ANN topologies and also to create artificial data to test ANN models.

2 Modelling Fermentation Processes

A fermentation process can be modelled by a set of Ordinary Differential Equations (ODE), which give the values of state variables at a particular instant. In these equations, there are a number of kinetic parameters to be determined, which is often a quite complex task. Several authors suggested that the best

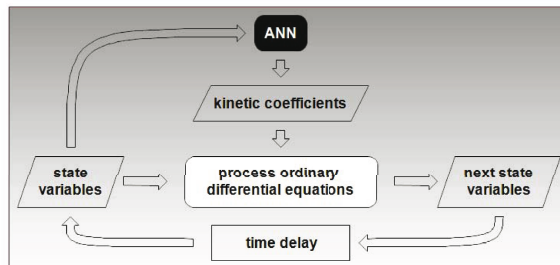


Fig. 1. Grey Box Model structure

approach is to use a hybrid model, where the contribution of various sources of information is merged for the construction of a single model [9, 6, 11, 17]. This approach was adopted as a basis to this work.

The kinetic component is, therefore, modelled using mechanisms that do not require an a priori knowledge of the process. These mechanisms are known as black box models, since only the inputs and the outputs of the system are known, while its internal processing is hidden since it is unknown or too complex. The data used for the construction of the kinetic component of the model can be obtained from real experiments. On the other hand, the structure of the ODEs is typically obtained from literature or biochemical knowledge. This component relies on mechanistic models, which are visible and can be interpreted (white box models). The merging of the two models in a single model leads to a hybrid structure, known as grey box models (Figure 1).

3 Description of the Computational Tools

3.1 Methodology

The software tools were all implemented using Java and an object-oriented approach. The main concerns in development were the modularity to allow code re-usage and the clear separation of the user interfaces and core functionality implementation. The tools are available at <http://darwin.di.uminho.pt/bionn>.

The computational tools developed in this work contemplate the white box and the grey box modelling strategies. The value of the state variables is always given by ODEs and the numerical simulation is performed using *ODEToJava*, a package of ordinary differential equation solvers [1]. The kinetic behavior can be modelled by two different strategies: heuristic approaches (white box) defined by some expert or taken from literature or the use of ANNs (black box approach). In this last case, a grey box modelling strategy is followed. The tools have been divided into two main groups: the ones related with white box models and the latter related to the grey box models.

3.2 White Box Interface

The White Box interface (Figure 2) includes a number of functionalities in the manipulation and simulation of white box models. In this area, it is possible to produce simulated data from white box heuristic models under different scenarios. The application considers a list of predefined fed-batch processes and models. It is possible to create new processes through the development of a Java source file that defines its behaviour, that is introduced and compiled at run-time. The simulated data generated in this area can be used to train and test ANNs in the grey box interfaces (next section), without the need to collect data from real cultures. The following list of functionalities is available:

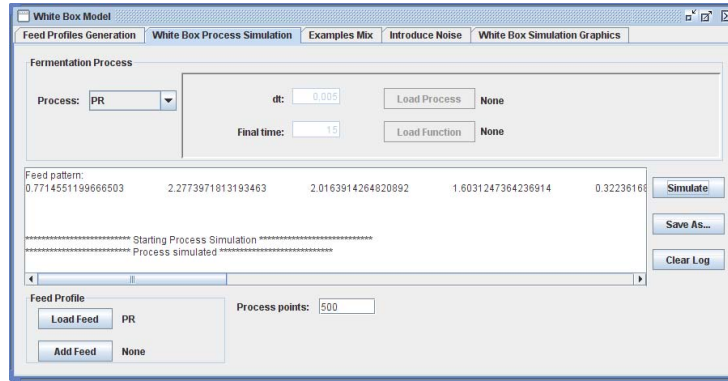


Fig. 2. Functionalities included in the White Box interface

Feeding Profiles Generation. The feeding profiles, i.e. the amount of substrate to provide to the culture at each time, represent the inputs of the simulation. Through this panel multiple profiles can be created with constant, linear, random, saw wave or sinus functions. Using different feeds to create data allows a broader spectrum of training examples.

White Box Process Simulation. This simulation uses heuristic models for generating the kinetics coefficients, which are then introduced into the ODE models. For each feeding profile, a distinct simulation can be conducted resulting in the values for state variables and kinetic parameters over time.

Mixing Examples. A number of files with simulations can be handled, and used to create new files through composition, cutting, interpolation or sampling data. These can create more comprehensive sets of examples or decrease the size of some collections without losing its generality.

Introducing Noise. The introduction of noise on data serves to create data sets that increase the generalization abilities of the ANN model. This can be useful when the set of available data is not rich enough to provide a good training process for the ANN.

White Box Simulation Graphs. The graphs allow for a better visualization of the data generated by the mathematical model. The preview is done in two separate graphs for the state variables and kinetics coefficients. It also allows visual comparisons between results of different simulations.

3.3 Grey Box Interface

The tools related to the grey box models include the training and evaluation of ANNs from experimental or simulated data and also the calculation of the model kinetic variables to integrate in the overall simulation (Figure 3). The following list of functionalities is available in this area:

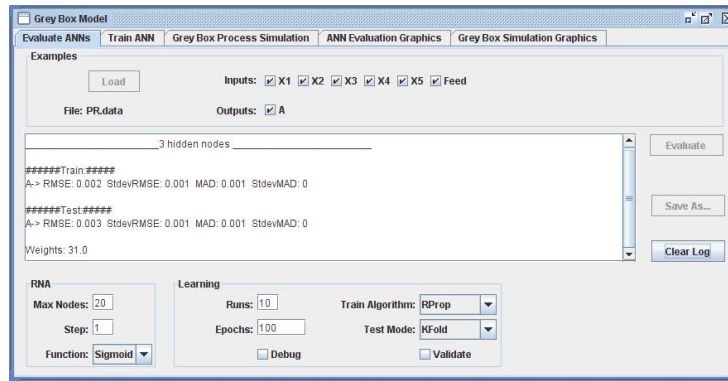


Fig. 3. Functionalities included in the Grey Box interface

Evaluate ANNs. This tool can assess various configurations of ANNs, varying the input and output variables, the number of nodes in the hidden layer (all ANNs are MLPs with one hidden layer, completely connected), training algorithm and the number of training epochs. Each model is evaluated by its training error and generalization error in a validation set. It is thus possible to determine which parameter combinations will lead to more accurate models.

ANN training. An ANN with a given configuration (typically the one that resulted from the above study) can be trained. It then can be used to predict the kinetic variables, integrated within the grey box model.

Grey Box Process Simulation. Enables the simulation of the fermentation process. Trained ANNs calculate the kinetic coefficients from state variables and ODEs simulate the mass-balance equations that determine the evolution of the values of the state variables.

ANN Evaluation Graphs. Show the errors from the evaluation of ANN configurations. They provide information on the settings, data and variables that were used in the training. The graphs allow a better interpretation of the error curves (both in training and validation sets).

Grey Box Simulation Graphs. Show data resulting from grey box model simulations. They show the evolution for each state variable and kinetic coefficient. It is also possible to compare two simulations and thus examine, for example, the difference between the curves obtained in the simulation model with grey box and the ones obtained with white box models.

4 Case Studies

4.1 PR Process

This process represents a *Saccharomyces cerevisiae* culture and was studied by Park and Ramirez (PR) [10]. The model equations (ODEs) are the following:

$$\frac{dx_1}{dt} = \frac{4.75A(x_2 - x_1)}{0.12 + A} - \frac{ux_1}{x_5} \quad (1)$$

$$\frac{dx_2}{dt} = \frac{x_3x_4e^{-5x_4}}{0.1 + x_4} - \frac{ux_2}{x_5} \quad (2)$$

$$\frac{dx_3}{dt} = (A - \frac{u}{x_5})x_3 \quad (3)$$

$$\frac{dx_4}{dt} = -7.3Ax_3 - \frac{u(x_4 - 20)}{x_5} \quad (4)$$

$$\frac{dx_5}{dt} = u \quad (5)$$

where x_1 , x_2 , x_3 and x_4 are the concentrations of secreted protein (units/L), total protein (units/l), cells (g/l) and substrate (g/l) respectively; x_5 is the fermenter's volume (l) and u the feed rate (l/h). The specific growth A (h^{-1}) is the only kinetic parameter and follows substrate inhibition kinetics:

$$A = \frac{21.87x_4}{(x_4 + 0.4)(x_4 + 62.5)} \quad (6)$$

4.2 Ecoli Process

This case study consists of an *Escherichia coli* culture. The kinetic behaviour is complex, presenting various coefficients and kinetic points of discontinuity which make this model a real challenge for the ANNs. The model is given by the following equations [12]:

$$\frac{dX}{dt} = (\mu_1 + \mu_2 + \mu_3)X - DX \quad (7)$$

$$\frac{dS}{dt} = q_sX + \frac{F_{in,S}S_{in}}{W} - DS \quad (8)$$

$$\frac{dA}{dt} = (k_3\mu_2 - k_4\mu_3)X - DA \quad (9)$$

$$\frac{dO}{dt} = (-k_5\mu_1 - k_6\mu_2 - k_7\mu_3)X + OTR - DO \quad (10)$$

$$\frac{dC}{dt} = (k_8\mu_1 + k_9\mu_2 + k_{10}\mu_3)X - CTR - DC \quad (11)$$

$$\frac{dW}{dt} \simeq F_{in,S} \quad (12)$$

being D the dilution rate, $F_{in,S}$ the substrate feeding rate (in kg/h), W the fermentation weight (in kg), OTR the oxygen transfer rate and CTR the carbon dioxide transfer rate.

The kinetic behavior is expressed in four variables: the rates μ_1 to μ_3 and q_s . A set of heuristic rules to calculate these values from state variables was defined in [12] and it is not shown here given its complexity. However, since it involves conditional branches it is discontinuous and also nonlinear.

4.3 Methodology

Using the previous white box models, a set of simulations was conducted with distinct feeding profiles. These were then be used to perform ANN evaluations and to test grey box models. To achieve this objective, it was necessary to follow a structured methodology to withdraw the best use of the case study. Overall, this approach has been structured into five steps: model selection; generation and selection of cases for training, selection of architecture and parameters; ANN training and simulation of the fermentation process.

5 Results

5.1 PR

For this case study, 22 distinct feed profiles were used: 4 random, 4 linear and 3 constants profiles and also included 11 optimum feed profiles (obtained by optimization with Evolutionary Algorithms [8]). All feedings have 31 values, which were linearly interpolated. Each of the profiles was used to generate a data file and all files were merged. Afterwards, the final training data set was obtained by sampling only 150 examples. This set of examples was used to test different ANN configurations by using 10-fold cross validation.

The first set of tests was conducted using only one state variable (1 ANN input). Then, other tests were performed in which all except one state variable were used. The results were compared, taking as a basis for comparison the case in which all state variables available were used as inputs. The comparative results showed that the best configuration would be the use of only the variable X_4 , a result that was expected given the expression used to compute A .

Regarding the number of hidden nodes, it was shown that the use of one intermediate node would already be viable to predict the kinetics accurately. However, 6 hidden nodes provided better generalization errors. So, an ANN with this configuration has been trained and used to simulate the process. In Figure 4 the results of the simulation to the process with this ANN are shown.

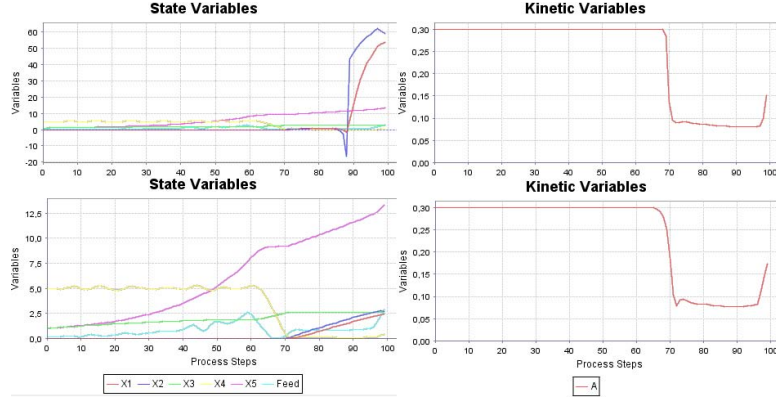


Fig. 4. Top: Simulation of the process PR (White Box). Bottom: Simulation of the process model with grey box model.

5.2 Ecoli

For this case study, 21 different feed profiles were used: 5 random, 5 linear, 4 constants and 7 optimum profiles. All feedings contained 26 values and each of the profiles was used to generate the data files. The method of selection of the state variables that influence the kinetic coefficients of the process was similar to the one used for PR. In this case, the conclusion was that the best configuration would be the use of variables OTR, CTR, X, A and F. This configuration was used to simulate the process using ANNs to compute the kinetic parameters, obtaining the result shown in Figure 5:

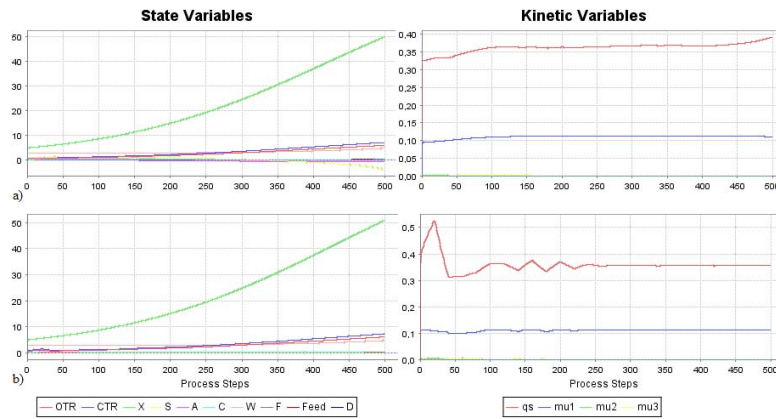


Fig. 5. Top: Simulation of the process Ecoli (White Box). Bottom: Simulation of the process with grey box model.

6 Conclusions and Further Work

A set of computational tools were developed to aid in the modelling of fed-batch fermentation processes. These can be used to handle data from real experiments or using simulated data. It is thus possible to shape a process without the need to know the mathematical description of its mechanism, since learning is made directly from examples created from experimental data, showing the potential of the application of ANNs in this type of problems.

It is possible to introduce new fermentation processes at run-time, enabling the application for a multitude of cases. It not only seeks to provide a solution to a specific case, but serves as a platform for a layman in ANNs to evaluate and use multiple models for each culture. This is also possible since user interfaces are implemented in a simple and intuitive way. The modularity of the application makes it scalable; the main modules can easily be used for future work. The core implementation of the functionalities is detached from the user interface, to make code understanding and re-use easier.

The implementation of several graphs allows making visual comparisons between results, providing easier analysis of the factors influencing the course of the process. Thus, it is possible to identify potential optimizations that can be made, such as to change the feed profile. These optimizations can be tested within the tool, thereby avoiding some real experiments that brings, necessarily, increased costs for research.

In the future, a major aim is to improve the capabilities of the application with new tools. A major concern is to create an interface to make the introduction of new processes easier, without having to write Java code.

Acknowledgments

This work was supported by the Portuguese FCT under project POSC/EIA/59899/2004.

References

1. Ascher, S., Ruuth.: Implicit-explicit runge-kutta methods for time-dependent partial differential equations. *Applied Numerical Mathematics* 25, 151–167 (1997)
2. Coulman, G.A., Stieber, R.W., Gerhardt, P.: *Dialysis Continuous Process for Ammonium-Lactate Fermentation of Whey: Mathematical Model and Computer Simulation*. American Society for Microbiology (1977)
3. Haykin, S.: *Neural Networks - A Comprehensive Foundation*, 2nd edn. Prentice-Hall, New Jersey (1999)
4. Jain, A.K., Mao, J., Mohiuddin, K.M.: *Artificial Neural Networks: A Tutorial*. IEE (1996)
5. Lednick, P., Mészáros, A.: Neural Network Modeling in Optimization of Continuous Fermentation Process. *Bioprocess Engineering* 18, 427–432 (1998)
6. Lee, D.S., Park, J.M.: Neural Network Modeling for On-line Estimation of Nutrient Dynamics in a Sequentially-operated Batch Reactor. *Journal of Biotechnology* 75, 229–239 (1999)

7. Levisauskas, D., Tekorius, T.: Model-Based Optimization of Fed-Batch Fermentation Processes Using Predetermined Type Feed-Rate Time Profiles. A Comparative Study. In: ITC (2005)
8. Mendes, R., Rocha, M., Rocha, I., Ferreira, E.C.: A Comparison of Algorithms for the Optimization of Fermentation Processes. In: Proceedings of the 2006 IEEE Conference on Evolutionary Computation, pp. 7371–7378. IEEE Computer Society Press, Los Alamitos (2006)
9. Oliveira, R.: Combining First Principles Modelling and Artificial Neural Networks: A General Framework. *Computers and Chemical Engineering* 28, 755–766 (2004)
10. Park, S., Ramirez, W.F.: Optimal Production of Secreted Protein in Fed-batch Reactors. *AIChE J.* 34(9), 1550–1558 (1988)
11. Peres, J., Oliveira, R., Azevedo, S.F.: Knowledge Based Modular Networks for Process Modelling and Control. *Computers and Chemical Engineering* 25, 783–791 (2001)
12. Rocha, I.: Model-based strategies for computer-aided operation of recombinant *E. coli* fermentation. PhD thesis, Universidade do Minho (2003)
13. Stanbury, P.F., Whitaker, A.: Principles of Fermentation Technology. Pergamon Press, Oxford (1984)
14. Taylor, B.J.: Methods and Procedures for the Verification and Validation of Artificial Neural Networks. Springer, Heidelberg (2006)
15. Veloso, A.C., Rocha, I., Ferreira, E.C.: On-Line Estimation of Biomass in an *E. Coli* Fed-Batch Fermentation. In: Enpromer (2005)
16. Zheng, Y., Gu, T.: Analytical Solutions to a Model for the Startup Period of Fixed-Bed Reactors. Elsevier Science (1996)
17. Zuo, K., Wu, W.T.: Semi-realtime Optimization and Control of a Fed-batch Fermentation System. *Computers and Chemical Engineering* 24, 1105–1109 (2000)